N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED IN THE INTEREST OF MAKING AVAILABLE AS MUCH INFORMATION AS POSSIBLE

NASA CR- 166655

MODIFICATIONS TO THE

FLEXIBLE SPACECRAFT DYNAMICS PROGRAM

Contract NAS 5-25678

(NASA-CR-166655) MODIFICATIONS TO THE FLEXIBLE SPACECRAFT DYNAMICS PROGRAM Final Report (Avco Systems Div., Wilmington, Mass.) 46 p HC A03/MF A01 CSCL 22B

N81-20166

Unclas

G3/18 19796

Final Report

15 March 1980

Prepared for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
GODDARD SPACE FLIGHT CENTER
Greenbelt, Maryland 20771

Prepared by

AVCO SYSTEMS DIVISION
201 Lowell Street
Wilmington, Massachusetts 01887

Table of Contents

1.0 SUMMARY

This report describes the modifications and additions made to the Flexible Spacecraft Dynamics (FSD) Program under contract NAS5-25678. The principal addition to the program was the capability to simulate the Dynamics Explorer-B Control System. The formulation for this addition is given in Section 2.0 of this report. The details of the modifications made to the FSD Program are presented in Section 3. Modifications to existing subroutines are briefly described and a detailed description of new subroutines is given. In addition, the program variables in new labelled COMMON blocks are described in detail. Section 4 gives a description of new and modified input and output for the FSD Program.

2.0 MATHEMATICAL FORMULATION FOR CONTROL SYSTEM SIMULATION

A block diagram of the system to be simulated is given in Figure 2-1. This block diagram is represented as a set of first order ordinary differential equations, which are integrated in parallel with the equations of motion for the rest of the spacecraft, using the same time step and integration algorithm. (The subroutine ADMIMP).

For the most part, the control components are linear dynamic systems. For such components, the stated transfer functions have been converted to state variable equations using standard techniques. This transformation, however, is not always unigue. Hence, it is necessary to state the exact form utilized in each case.

In the equations following, the subscripts 1, 2 etc. are used primarily for convenience. However, the ordering of variables is the same as in program code. Hence a fourth order model with state variables $x_1 \ldots x_4$ may appear in program code as $x_6 \ldots x_9$. The actual subscripts used in program code are given in the section on program inputs. Also, in this section, the symbols u and y denote (respectively) the input to and output from the given transfer function. In the system simulation, the blocks are coupled together.

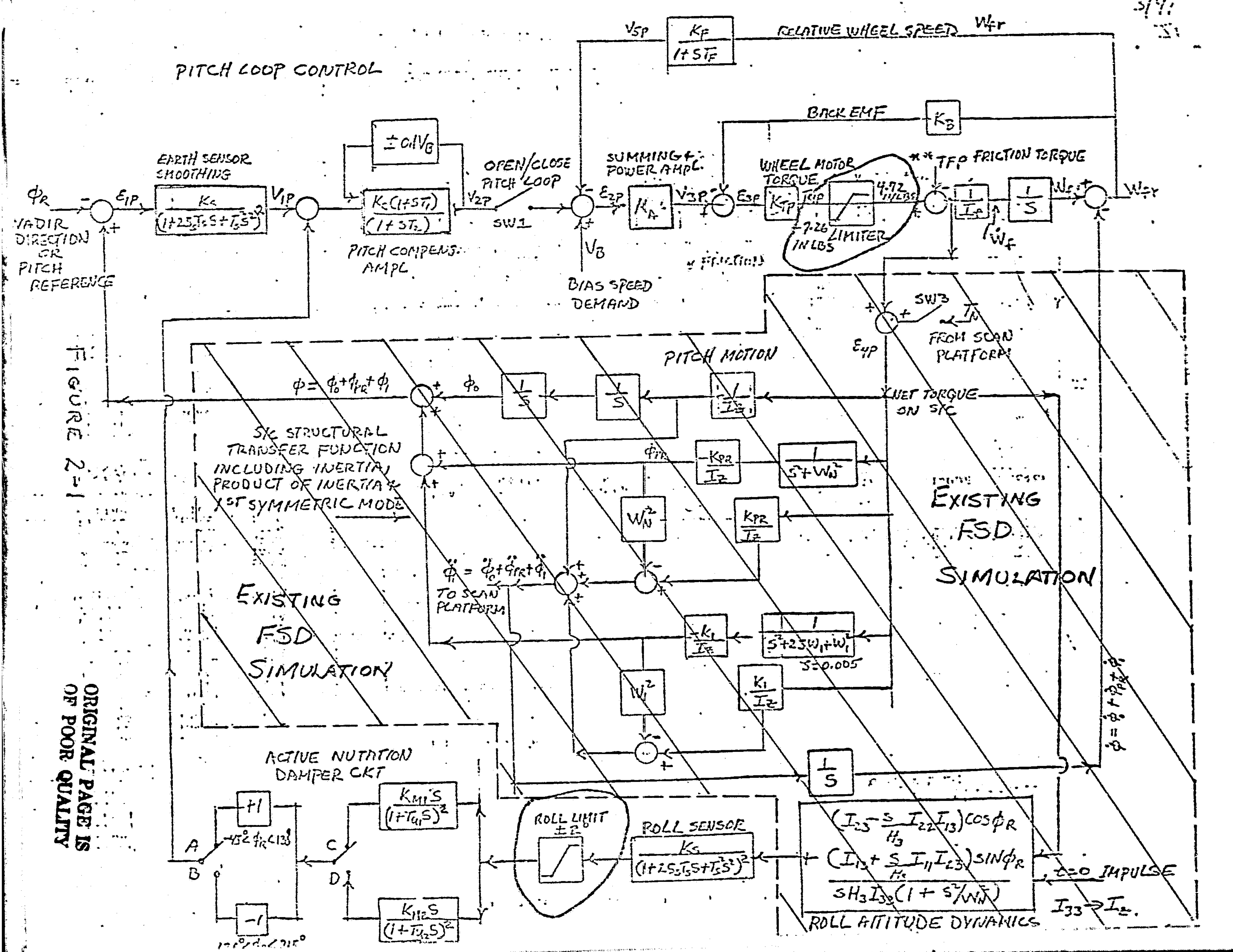
PITCH LOOP CONTROL
RELATIVE WHEEL SPEED
BACK EMF
EARTH SENSOR SMOOTHING
OPEN/CLOSE PITCH LOOP
SW1
SUMMING & POWER AMPL.
WHEEL MOTOR TORQUE
LIMITER
IN LBS
FRICTION TORQUE
PITCH COMPENS. AMPL
BIAS SPEED DEMAND
NADIR DIRECTION OR PITCH REFERENCE
SW3
FROM SCAN PLATFORM
PITCH MOTION
NET TORQUE ON S/C
S/C STRUCTURAL TRANSFER FUNCTION INCLUDING INERTIA, PRODUCT OF INERTIA & 1ST SYMMETRIC MODE
TO SCAN PLATFORM
EXISTING FSD SIMULATION
EXISTING FSD SIMULATION
ACTIVE NUTATION DAMPER CKT
ROLL LIMIT
ROLL SENSOR
ROLL ATTITUDE DYNAMICS
IMPULSE

FIGURE 2-1

Pitch or roll sensor (2nd order model)

The transfer function

$$\frac{K_s}{(1+sT_s)^2}$$

is represented as

$$\frac{d}{dt}\begin{Bmatrix} X_1 \\ X_2 \end{Bmatrix} = \begin{bmatrix} -\frac{1}{T_s} & \frac{K_s}{T_s} \\ 0 & -\frac{1}{T_s} \end{bmatrix} \begin{Bmatrix} X_1 \\ X_2 \end{Bmatrix} + \begin{Bmatrix} 0 \\ \frac{1}{T_s} \end{Bmatrix} u$$

$$Y = X_1$$

Pitch or roll sensor (4th order model)

The transfer function

$$\frac{K_s}{(1+2\zeta_s T_s s + T_s^2 s^2)^2}$$

is represented as

$$\frac{d}{dt}\begin{Bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{Bmatrix} = \begin{bmatrix} 0 & \frac{1}{T_s} & 0 & 0 \\ -\frac{1}{T_s} & -2\frac{\zeta_s}{T_s} & \frac{K_s}{T_s} & 0 \\ 0 & 0 & 0 & \frac{1}{T_s} \\ 0 & 0 & -\frac{1}{T_s} & -2\frac{\zeta_s}{T_s} \end{bmatrix} \begin{Bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{Bmatrix} + \begin{Bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{T_s} \end{Bmatrix} u$$

$$Y = X_1$$

Nutation Damper Phase Shift Circuit

e transfer function

$$\frac{K_M s}{(1+sT_M)^2}$$

is represented as

$$\frac{d}{dt}\begin{Bmatrix} X_1 \\ X_2 \end{Bmatrix} = \begin{bmatrix} -\frac{1}{T_M} & \frac{1}{T_M} \\ 0 & -\frac{1}{T_M} \end{bmatrix}\begin{Bmatrix} X_1 \\ X_2 \end{Bmatrix} + \begin{Bmatrix} \frac{K_M}{T_M^2} \\ -\frac{K_M}{T_M^2} \end{Bmatrix}$$

$Y = -X_1$ (the minus sign includes the sign inversion corresponding to path A in Figure 2-I)

The inversion of sign at various roll angles and the switch between primary and secondary damper may be simulated by changing the input parameters.

Tachometer

The transfer function $\dfrac{K_F}{1 + sT_F}$

is represented as

$$\frac{d}{dt}X_1 = -\frac{1}{T_F}X_1 + \frac{K_F}{T_F}u$$

$$Y = X_1$$

For the reference control system, the tachometer significantly faster than the other dynamic elements. This causes the tachometer equation to dominate the time step control in numerical integration, which having little effect on system performance. Replacing the above transfer function with the static operator $Y = K_F u$ permits a significant reduction in program execution time. This alt[illegible]ative model is optionally available in the modified program.

Pitch Compensation Amplifier

In unsaturated operation, the transfer function

$$\frac{K_c(1 + sT_1)}{(1 + sT_2)}$$

is represented as

$$\frac{d}{dt} x_1 = -\frac{1}{T_2} Y + \frac{K_c}{T_2} u$$

$$Y = x_1 + K_c \frac{T_1}{T_2} u$$

Saturation occurs if $|Y| > V_{LIM}$

If this occurs, Y is replaced by V_{lim} sign(Y).

3.0 PROGRAM ADDITIONS AND MODIFICATIONS

The additions and modifications to the FSD Program accomplished under this contract are described below.

3.1 Summary of Modifications by Task

Task 1 DYNAMICS EXPLORER-B CONTROL SYSTEM

A simulation capability for the Dynamics Explo[illegible] B Control System has been added to the FSD Program.

1.1 Sensor Module

The sensor module converts the state vector from the FSD simulation to the appropriate attitude angles (yaw, roll, and pitch) for sensor output. The pitch angle is compared to command pitch attitude and the resulting angular roll and pitch error are passed through either second order or fourth order lag circuits to the control system module. Provision has been made for adding bias errors and noise to the attitude angles obtained from the FSD state vector.

1.2 Control System Module

The control system module provides the driving torque to a variable speed momentum wheel. The driving torque is determined from a simulation of the momentum wheel motor. The motor input is the sum of a constant bias speed voltage and a voltage derived from pitch and roll attitude error signals. The roll error signal is used for nutation damping. The pitch error signal is used to maintain a constant pitch reference direction.

The roll error signal is clipped and passed through a phase shifting circuit and the output is added to the pitch error signal. The resultant sum is passed through a compensation amplifier with voltage limiting.

The print and plot subroutines of the FSD Program have been modified to permit printing and plotting of control system variables.

The control sys[illegible] transfer functions are integrated as added state variabl[illegible] the FSD integrator ADMIMP.

The DE-B control system simulation has been checked out with rigid body examples supplied by GSFC.

Task 2 EXTERNAL MOMENTS

Provision has been made to print and plot the sum of all external moments acting on the system. This sum includes gravity gradient, solar pressure, aerodynamic pressure, magnetic, and all thruster related control torques.

Task 3 NOISE GENERATOR

A Gaussian noise generator has been included so that the effect of noise on control system performance can be evaluated. This generator is based on subroutine GAUSS taken from the IBM System/360 scientific subroutine package.

Task 4 THRUSTER OPTION

The thruster option has been modified so that two thrust pulses can be applied to the spacecraft during a rotation period or time interval. The parameters defining the pulse characteristics are independent.

Task 5 FAST FOURIER TRANSFORM OPTION

The Fast Fourier Transform Option has been modified so that up to four frequencies and amplitudes can be printed.

Task 6 PROGRAM DISCREPANCIES

6.1 <u>Spin Axis Moment Option</u>

The spin axis moment subroutines have been corrected.

6.2 <u>Deployment Acceleration</u>

The deployment acceleration has been reset to zero at the end of each deployment phase.

6.3 <u>Excessive IØ Time</u>

The overlay structure has been adjusted so that overlay IØ time will be acceptable.

3.2 DESCRIPTION OF SUBROUTINE MODIFICATIONS

The subroutine modifications are briefly described below.

ADMIMP

Eliminate initialization logic for integrator messages (moved to subroutine BØUNDS). Eliminate common block INTEGF. Call subroutine RWHOUT after every integration step.

BØUNDS

Eliminate arguments in calling sequence. Establish default (large) values for bounds not specified via input. Initialize integrator message variables.

CSD

Logic changes so that the number of frequencies is specified by the input word ICSD.

ECHØA

Echo printout was added for control system parameters. The output for the thruster option was changed to reflect two pulses per spin period.

GPRINT

Call to DEBØUT was inserted to provide for printed output of control system behavior.

MAIN

Modify calls to subroutine NUM, BOUNDS to eliminate arguement. Call subroutine CSIC, VDIC. Print program version number after every call to READIN. Calls subroutine PDTAPR instead of DTAPRE.

Arrays DER, DEP now in labelled common ADSTAT. References common block ICNTRL. Eliminate common blocks CNBODY, VECTRS.

NUM

Eliminate arguments in call sequence. Calls NUMCSE if control simulation is specified.

ØRBUPD

Modification for counting of pulses under conditions of two pulses per spin period.

PULSER

Required changes to allow two pulses per spin period.

PULSUN

Required changes to allow two pulses per spin period.

RD478

Setup calls added for control system simulation input.

READTH

Setup calls and CØMMON/THRUST/changed for two pulses per spin period.

SH

Revise the direction cosine matrix normalization. Provide transfer of rigid body angular rates from DEPEND array to OMEG vector (in common block RPOOL1).

TØTIMP

Required changes to allow two pulses per spin period.

TTRUST

Required changes to allow two pulses per spin period.

WHEELS

Added control equations for closed loop control of momentum wheel speed.

WRTPLT

Plotting capability added for body momentum vector components, external moments, and control system state vector.

3.3 DESCRIPTION OF NEW SUBROUTINES

The new subroutines written for the DE-B control system simulation are described in the folliwing section.

SUBROUTINE NAME: CSIC

LANGUAGE: FORTRAN IV

CALLING SEQUENCE: CALL CSIC

PURPOSE: Enter initial conditions on control system state variables into system state vector.

COMMONS USED: ADSTAT, CSTAT, JCNTRL, IMAIN1

CALLED BY: MAIN

SUBROUTINES CALLED: NONE

GLOSSARY OF VARIABLES: See common block descriptions.

SUBROUTINE NAME: DEBANG

LANGUAGE: FORTRAN IV

CALLING SEQUENCE: CALL DEBANG(YAW, ROLL, PITCH)

PURPOSE: Calculates the attitude of the spacecraft with respect to the local vertical frame. The sequence used in going from the local vertical to the body frame is in the order yaw(3), roll(1) pitch(2).

LABELLED COMMONS USED: RPOOL1, VECTRS

CALLED BY: WHEELS

SUBROUTINES CALLED: DSQRT

GLOSSARY OF VARIABLES: OUTPUTS

YAW	ROTATION ABOUT THE THREE AXIS.
ROLL	ROTATION ABOUT THE ONE AXIS.
PITCH	ROTATION ABOUT THE TWO AXIS.

SUBROUTINE NAME: DEBØUT

LANGUAGE: FORTRAN IV

CALLING SEQUENCE: CALL DEBØUT

PURPOSE: Provides the printed output for the control system using the subroutine SET.

LABELLED COMMONS USED: CSTAT

CALLED BY: GPRINT

SUBROUTINES CALLED: SET

GLOSSARY OF VARIABLES: See common block description.

SUBROUTINE NAME: DEREQ

LANGUAGE: FORTRAN IV

CALLING SEQUENCE: CALL DEREQ

PURPOSE: Executive routine for derivative computation

COMMONS USED: See former DEREQ1

CALLED BY: DEREQ1

SUBROUTINES CALLED: See former DEREQ1

GLOSSARY OF VARIABLES: See former DEREQ.

ICATIONS: Eliminate arguments in calling sequence. DEREQ contains coding formerly in subroutine DEREQ1.

SUBROUTINE NAME: DEREQ1

LANGUAGE: FORTRAN IV

CALLING SEQUENCE: Call DEREQ1 (QDPEND, TIME, QDERIV)

PURPOSE: Intermediary between subroutines ADMIMP and DEREQ DEREQ1 takes state variables passed as an argument and places them in labelled common VARBLS. After calling DEREQ, this program also collects the calculated derivatives and returns them via the argument list.

COMMONS USED: IMAIN1, RPOOL1, VARBLS

CALLED BY: ADMIMP

SUBROUTINES CALLED: DEREQ

GLOSSARY OF VARIABLES:

INPUTS	
TIME	Time in seconds from ADMIMP
NUMEQS	Number of state variables
QDEPEND	Values of state variables received from ADMIMP
OUTPUTS	
TIM1	Time in seconds, passed to all other programs
DEPEND	Values of state variables passed to all other programs
DERIV	Values of state varible derivatives received from computation programs
QDERIV	Values of state variable derivatives returned to ADMIMP

SUBROUTINE NAME: NUMCSE

LANGUAGE: FORTRAN IV

CALLING SEQUENCE: CALL NUMCSE

PURPOSE: Determine number of control equations and defines mapping between system and control state vectors.

COMMONS USED: ICNTRL, IMAIN1, JCNTRL

CALLED BY: NUM

SUBROUTINES CALLED: NONE

GLOSSARY OF VARIABLES: See common block descriptions

SUBROUTINE NAME; PDTAPR

LANGUAGE: FORTRAN IV

CALLING SEQUENCE: Call PDTAPR

PURPOSE: Begin reading ephemeris tape

COMMONS USED: CNBODY, IMAINS, TJAN1, VECTRS

CALLED BY: MAIN

SUBROUTINES CALLED: DTAPRE, IBCOM#

GLOSSARY OF VARIABLES: See prior MAIN

SUBROUTINE NAME: RWHØUT

LANGUAGE: FORTRAN IV

CALLING SEQUENCE: Call RWHØUT

PURPOSE: Obtain trajectory data for external plot programs
Writes to unit 50 when called. Called from ADMIMP
after every time step.

COMMONS USED: CSTVAL, VARBLS

CALLED BY: ADMIMP

SUBROUTINES CALLED: IBCOM#

GLOSSARY OF VARIABLES:

TTST	Time in seconds
DEPEND	State variable values
DERIV	State variable derivatives

SUBROUTINE NAME: VDIC

LANGUAGE: FORTRAN IV

CALLING SEQUENCE: Call VDIC

PURPOSE: To compute initial conditions for viscous damper and insert into system state vector.

COMMONS USED: ADSTAT, CONSTS, IMAIN1, RPOOL1, RVISCS

CALLED BY: MAIN

SUBROUTINES CALLED: NONE

GLOSSARY OF VARIABLES: See former MAIN program

SUBROUTINE NAME: EXPN

LANGUAGE: FORTRAN IV

CALLING SEQUENCE: Call EXPN (T, TLAST, TI VLAST, VI SSN, TCOR, OUT)

PURPOSE: To generate noise channel-outputs for control system simulation.

LABELLED COMMONS USED: ICSADM

CALLED BY: WHEELS

SUBROUTINES CALLED: GAUSS

GLOSSARY OF VARIABLES:

T	Current integrator time
TLAST	Time at last call to EXPN
TI	Time at last successful integration time
VLAST(I)	Noise channel output at time TLAST
VI(I)	Noise channel output at time TI
SSN(I)	Noise channel sigma
TCOR(I)	Noise channel lag
OUT(I)	Noise channel output at time T

3.4 DESCRIPTION OF NEW AND MODIFIED LABELLED COMMON

The new labelled common blocks are described in the following section. The only modification of labelled common is the COMMON/THRUST/. This common was changed to reflect the specification of two pulses per spin period.

COMMON BLOCK NAME:

COMMON/THRUST/TV(3,2), TLØC(3 2), TTIM(4,2),TPAR(4,2),REF(2)

The difinitions of these symbols are given in the input description of Section 4.2

COMMON BLOCK NAME: COMMON/ADSTAT/DER(150), DEP(150)

USED IN SUBROUTINES: MAIN, VDIC, CSIC

PURPOSE: The DER and DEP arrays were formerly local to the MAIN Program. They have been placed in common so that initial condition can be loaded by VDIC, CSIC and future programs of the same general type.

VARIABLES:

FORTRAN NAME	TYPE	UNITS	DESCRIPTION
DER(150)	R*8	variable	State variables
DEP(150)	R*8	variable	State variable derivative

COMMON BLOCK NAME: COMMON/CSBNDS/CSUP(20), CSDN(20), GNIC(10)

USED IN SUBROUTINES: CSIC, RD478

PURPOSE: To transfer integration bounds and noise model initial conditions.

VARIABLES:

FORTRAN NAME	TYPE	UNITS	DESCRIPTION
CSUP (20)	R*8	varies	Upper integration bound for control system state vector
CSDN (20)	R*8	varies	Lower integration bound for control system state vector
GNIC (10)	R*8	varies	Noise channel initial conditions

COMMON BLOCK NAME: COMMON/CNOISE/VNS2(10), VNSI(10), VNSN(10),

T1 , T3

USED IN SUBROUTINES: CSIC, WHEELS

PURPOSE: Transfer noise channel data.

VARIABLES:

FORTRAN	TYPE	UNITS	DESCRIPTION
VNS2(10)	R*8	varies	Noise channel outputs for last call to EXPN.
VNS1(10)	R*8	varies	Noise channel outputs for last successful time step for integrator.
VNSN(10)	R*8	varies	Noise channel outputs for current call to EXPN.
T1	R*8	sec.	Time of last successful integration
T2	R*8	sec.	Time of last call.

COMMON BLOCK NAME: COMMON/CSTAT/X(20), XDOT(20), CPARM(43)

OR

COMMON/CSTAT/SVCS(20), SVDCS(20), CPARM(43)

USED IN SUBROUTINES: RD478, WHEELS, CSIC, WRTPLT, DEBØUT

PURPOSE: Carries control system state vector, its derivative and control system parameters.

VARIABLES:

FORTRAN NAME	TYPE	UNITS	DESCRIPTION
X	R*8	see inputs descriptions	Control system state vector.
XDOT	R*8		Derivative of control system state vector.
CPARM	R*8		Control system parameters.

COMMON BLOCK NAME: COMMON/ICNTRL/KNTRL(10)

USED IN SUBROUTINES: MAIN, RD478, WHEELS, NUM, NUMSCE, WRTPLT

PURPOSE: Specifics control system options.

VARIABLES:

FORTRAN NAME	TYPE	UNITS	DESCRIPTION
KNTRL	I*4		See input descriptions.

COMMON BLOCK NAME: COMMON/ICSADM/LSAVE, IRAND, NCHAN

USED IN SUBROUTINES: ADMIMP, CSIC, EXPN, WHEELS

PURPOSE: To transfer noise generator control words

VARIABLES:

FORTRAN NAME	TYPE	UNITS	DESCRIPTION
LSAVE	I*4	N.A.	Integrator logic integer
IRAND	I*4	N.A.	Random number integer
NCHAN	I*4	N.A.	Number of noise channels

COMMON BLOCK NAME: COMMON/JCNTRL/NCNTRL, MCNTRL, MAPCNT(20)

USED IN SUBROUTINES: CSIC, WHEELS, NUMCSE

PURPOSE:

VARIABLES:

FORTRAN NAME	TYPE	UNIT	DESCRIPTION
NCNTRL	I*4		Number of control equations
MCNTRL	I*4		Index of state variable immediately preceeding 1ST control state variable
MAPCNT(20)	I*4		Defines mapping between system state vector and control state vector

COMMON BLOCK NAME: COMMON/SCSTAT/SSVCS(20)

USED IN SUBROUTINES: CSIC

PURPOSE: To save control system state vector initial conditions for stacked cases.

VARIABLES:

FORTRAN NAME	TYPE	UNITS	DESCRIPTION
SSVCS(20)	R*8	varies	Initial conditions for control system state vector

COMMON BLOCK NAME: COMMON/VERS/VNO

USED IN SUBROUTINES: MAIN, BVERS, (BLOCK DATA)

PURPOSE: Identifies current program version used to determine modification level.

VARIABLES:

FORTRAN NAME	TYPE	UNITS	DESCRIPTION
VNO	R*8		8 character version identification

4.0 DESCRIPTION OF NEW AND MODIFIED INPUT/OUTPUT

The new and modified input symbols and output capabilities for the FSD program are detailed in the following section.

4.1 DYNAMICS EXPLOROR-B SIMULATION INPUT/OUTPUT

The input symbols for the DE-B control system simulation are described below:

INPUT SYMBOL	TYPE	PRESET VALUE	DESCRIPTION OF USE
KNTRL(10)	I*4	0	Vector of control integers for DE-B control system simulation

KNTRL ADDRESS	DESCRIPTION
(1)	KNTRL(1)=0 No control system KNTRL(1)=1 Control system with second order sensors KNTRL(1)=2 Control system with fourth order sensors
(2)	KNTRL(2)=0 No nutation damper KNTRL(2)=1 Primary damper circuit KNTRL(2)=2 Offset pointing damper circuit
(3)	KNTRL(3)=0 No KNTRL(3)=1 First order tachometer
(4)-(8)	NOT USED
(9)	KNTRL(9)=I I is starting integer for noise generator. I <u>must be odd</u> and should have 6 or 7 digits
(10)	KNTRL(10)=0 No noise channels KNTRL(10)=3 Noise generated for sensors & bial voltage. Use only 0 or 3

INPUT SYMBOL	TYPE	PRESET VALUE	DESCRIPTION
CPARM(43)	R*8	(1-30) 0.000 (31-40) 1.000 (41-43) 0.000	Control system parameters for DE-B control system simulation

CPARM ADDRESS	MATH SYMBOL	UNITS	DESCRIPTION
(1)	τ_s	sec.	Sensor time constant
(2)	τ_1	sec.	Lead term in pitch compensation
(3)	τ_2	sec.	Lag term in pitch compensation
(4)	τ_F	sec.	Tachometer time constant
(5)	K_s	volts/rad.	sensor gain
(6)	K_c	volts/volt	Pitch amplifier gain
(7)	K_a	volts/volt	Power amplifier gain
(8)	K_f	volts/(rad/sec)	Tachometer gain
(9)	K_b	volts/(rad/sec)	Motor back EMF constant
(10)	K_t	ft.lbs/volt	Morot torque constant
(11)			Not used for input
(12)	V_{lim}	volts	Voltage limit in compensation amplifier
(13)	$K_{\mu 1}$	volts/volt	Gain in primary damper circuit
(14)	$\tau_{\mu 1}$	sec.	Time constant in primary damper circuit
(15)	V_b	volts	Bias boltage
(16)	$\mathcal{T}_{co}$	ft-lbs	Coulomb friction torque
(17)	Ω_{min} $\mathcal{T}_c = \mathcal{T}_{co}\frac{1}{\Omega_{min}+\lvert\Omega_w\rvert}$	rad/sec	Test relative wheel speed to avoid coulomb friction touque discontinuity at zero speed

CPARM ADDRESS	MATH SYMBOL	UNITS	DESCRIPTION
(18)	ζ_s	volts/volts	Fourth order sensor
(19)	$K_{\mu 2}$	sec	Gain in offset pointing damper circuit
(20)	$\tau_{\mu 2}$	N.D.	Time constant in offset pointing damper circuit
(21)		N.D.	Sign of damper circuit output set 10 1.0 or -1.0
(22)		volts	Roll sensor output limit
(23)		ft.-lbs.	Motor torque output upper limit
(24)		ft.-lbs.	Motor torque output lower limit
(25-30)			Not used
(31-33)			Noise model SIGMA for pitch, roll and voltage bias respectively
(34-35)			Not used
(36-38)			Noise model LAG for pitch, roll and voltage bias respectively
(39-40)			Not used
(41)		rad.	Pitch sensor bias
(42)		rad.	Roll sensor bias
(43)			Not used

INPUT SYMBOL	TYPE	PRESET VALUE	DESCRIPTION
SVCS(20)	R*8	0.000	Initial conditions for control system state vector

SVCS ADDRESS	UNITS	DESCRIPTION
(1)	volts	Pitch sensor output
(2-4)	volts	Pitch sensor dynamics
(5)		Not used
(6)	volts	Roll sensor output
(7-9)	volts	Roll sensor dynamics
(10)		Not used
(11)	volts	State variable for pitch compensation amplifier
(12)		Not used
(13)	volts	Tachometer output
(14)		Not used
(15)	rad/sec	Whe el speed
(16-18)		Not used
(19)	volts	Nutation damper
(20)	volts	Nutation damper

INPUT SYMBOL	TYPE	PRESET VALUE	DESCRIPTION
GNIC(10)	R*8	0.0D0	Initial conditions for noise model channels. GNIC(1) Pitch channel GNIC(2) Roll channel GNIC(3) Bias voltage channel GNIC(4-10) Not used
CSUP(20)	R*8	1.0D-2	Upper bound on difference between predicted and corrected control system state vector. Location in CSUP corresponds to the location of the variable in the state vector initial condition array SVCS
CSDN(20)	R*8	1.0D-4	Lower bound on difference between predicted and corrected control system state vector. Location in CSDN corresponds to the location of the variable in the state vector initial condition array SVCS

In addition to the new input described above, the following must be given:

IWHEEL	1	Forces call to momentum wheel subroutine (WHEELS)
XMOMIN(2)	___	Inertia of momentum wheel ($slug/ft^2$)

It is recommended that integration control bounds be given for critical control variables to prevent integration time step from exceeding the time constants of the closed-loop system. For example, if fastest component in the control system has a time constant of .1 sec, it is unreasonable to expect accurate simulation results with a larger time step. Setting the integration bounds to some reasonable (small) fraction of the nominal value (eg 10^{-2} to 10^{-4}) will ensure that the integration errors are of the same order (eg 2 to 4 place accuracy). The time stop corresponding to this level of accuracy will be determined internally.

The output for the DE-B control system simulation includes both printed data and plots. The printed output is as follows:

OUTPUT HEADING	UNITS	DESCRIPTION
PTCH Out		Pitch channel sensor output
ROLL Out		Roll channel sensor output
COMP Out		Output of compensation amplifier
TACH Out		Output of tachometer
WHL SPD	rad/sec	Momentum wheel speed
NUTD Out		Nutation damper phase shift circuit output

The output available for plotting is the entire state vector for the control system. The locations and definitions of these variables are as follows:

KPLOTS ADDRESS	UNITS	DESCRIPTION
(216)		Pitch sensor output
(217-219)		Pitch sensor dynamics
(220)		Not used
(221)		Roll sensor output
(222-224)		Roll sensor dynamics
(225)		Not used
(226)		Output of pitch compensation amplifier
(227)		Not used
(228)		Tachometer output

ADDRESS	UNITS	DESCRIPTION
(229)		Not used
(230)		Momentum wheel speed
(231-233)		Not used
(234)		Nutation damper phase shift dynamics
(235)		Nutation damper phase shift output

4.2 MODIFIED THRUSTER OPTION INPUT

<u>THRUST LOADING OPTION</u>

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	UNITS
IPULSE	N.A.	Control word to activ te IPULSE=0 No thrusting IPULSE=1 Apply thrust once IPULSE $\geq$1 apply thrust IPULSE times (only if ISPLSE $\neq$ 0.)	N.D.
ISPLSE	N.A.	Control word to activate sun crossing time to start the thrusting ISPLSE=0 Sun crossing not used ISPLSE=1 Sun crossing used	N.D.
ISPNP	N.A.	Control word to print out the orbit update message (only if ISPLSE=1, IPULSE 1) i.e., if ISPNP=5, the orbit update message will be printed at every 5th pusle	
IPLPRP	N.A.	Control word for number of thrust pulses per spin record IPLPRP=1 One pulse IPLPRP=2 Two pulses Only one or two pulses are allowed. (Preset =1)	N.D.

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	UNITS
TVECTR(3,2)	(D_V)	Unit vector defining the direction of the force applied to the body due to thrusting. This vector is defined in the body frame. (preset= 0.0, 0.0, 1.0)	N.D.
TLOCAT(3,2)	$(_T)$	Location in the body frame of the point application of the force due to thrusting. (preset=0.0)	feet
TTIMES(4,2)		Times to define thrust variation measured from the problem starting time	sec
	t_1	TTIMES(1,I) Start of pulse	
	t_2	TTIMES(2,I) End of exponential rise	
	t_3	TTIMES(3,I) End of linear thrust	
	t_4	TTIMES(4,I) End of pulse I=1 or 2	
TPARAM(4,2)		Parameters to define thrust variation	
	A	TPARAM(1,I) Coefficient during exponential rise	lb
	B	TPARAM(2,I) Exponential decay constant durint dxponential rise	sec^{-1}
	C	TPARAM(3,I) Coefficient for linear slope	lb/sec
	D	TPARAM(4,I) Exponential decay constant during exponential decay	sec^{-1}

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	UNITS
REFANG(2)	A_R	Angular delay from the Y_1 axis crossing the sun line to the pulse	deg

4.3 MODIFIED FFT INPUT AND ADDITIONAL PLOT CAPABILITY

Fast Fourier Transform (FFT) Analysis

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	UNITS
ICSD	N.A.	Control word to activate the FFT analyses subroutine. ICSD=0 No FFT analysis ICSD=N FFT analysis activated (preset=0) The integer N requests that N frequencies be extracted from the selected KPLOTS data sets	N.D.

NOTE: The value of N should not exceed the number of frequencies that can reasonably be expected to exist in the data. The range of permissible values for N are from 1 to 10.

$$1 \leq N \leq 10$$

KPLOTS ARRAY ADDRESS I	DESCRIPTION	FORTRAN SYMBOL	UNITS	NOTE
210	External moment about 1 body axis	MOMENT 1	ft-lbs	
211	External moment about 2 body axis	MOMENT 2	ft-lbs	
212	External moment about 3 body axis	MOMENT 3	ft-lbs	
213	Component of angular momentum on 1 body axis	HBODY 1	slug-ft^2/sec	IHCALC 1
214	Componant of angular momentun on 2 body axis	HBODY 2	slug-ft^2/sec	IHCALC 1
215	Componant of angular momentum of 3 body axis	HBODY 3	slug/ft^2/sec	IHCALC 1
216 → 235	Control system state vector (See Section 4.1)	N.A.	varies	IWHEEL 1 and KNTRL(1) 1 or 2